

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Patent Application for

Method, Device and Computer Program Product
for Image Stabilization Using Color Matching

Invention of:

Susan Heath Calvin Fletcher
333 Ruby Drive
Huntsville, AL 35811

Gene Arthur Grindstaff
2202 Essex Drive SW
Decatur, AL 35603

Therman Ward McKay III
6603 Willow Springs Blvd.
Huntsville, Alabama 35806

Attorney docket number:
1247/A70

Attorneys:
Bromberg & Sunstein LLP
125 Summer Street
Boston, MA 02110-1618
Tel: (617) 443-9292
Fax: (617) 443-0004

Method, Device and Computer Program Product
for Image Stabilization Using Color Matching

5

Priority

This application claims priority from U.S. Provisional Application No. 60/318,164,
filed September 7, 2001 entitled "Method, Device and Computer Program Product for Analysis
10 of Video Images", the full disclosure of which is hereby incorporated by reference herein in its
entirety and the present application also claims priority from U.S. Application No. U.S. Patent
Application 10/237,167 entitled "Method, Device and Computer Program for Image
Stabilization Using Color Matching" filed on September 6, 2002 which is incorporated by
reference herein in its entirety.

15

Technical Field

The present invention relates to analysis of video images and more specifically
to image stabilization of video images.

20

Background

Video stabilization is known within the prior art to minimize image jitter due to
camera movement through either mechanical feedback systems or through digital signal
processing. The digital signal processing techniques of the prior art are complicated and
often are based upon motion estimation and vector analysis. These techniques are
25 designed so as to be minimally intrusive to movement within the image and assess
movement of blocks of pixels in order to isolate camera movement jitter.

When objects move within captured video images as opposed to the camera
moving details of the moving object being captured often are hard to discern. For
example, a license plate of a moving car is often hard to read when the captured video
30 images are displayed due to the great amount of motion from frame to frame of the car.
In such a situation, an accounting for the motion of the car would enhance the ability to
read the license plate.

The traditional image stabilization techniques as discussed above do not provide

a way to account for motion such as the motion of a car in order to make a license plate more legible because the techniques are designed to allow such motion to be visible within the sequence of video images while only removing movement due to the movement of the camera.

5

Summary of the Invention

10 In one embodiment of the invention, a method for image stabilization of at least two digital images is disclosed. In such an embodiment, a first digital image having a plurality of pixels and at least a second image having a plurality of pixels are provided. Each pixel has an associated address for display and is representative of a color. Either a user of the system provides a color match range or the system provides a predetermined color matched range. A pixel is selected within the first digital image. Preferably the pixel is one that represents an item within the image that is either blurry due to movement of the item or appears jittery due to camera movement. Due to the vast number of available colors a pixel selected in a first image can be matched to a pixel in a second image within a range. The range allows for compensation due to lighting changes. Once the pixel in the first image is selected it can be compared to all pixels within the second image. Each pixel within the second image that is within the color match range is saved and the closest pixel color to the pixel from the first image is selected. The address of the pixels in the second image are then readdressed such that the address of the pixel located in the second image that has the closest color to the pixel in the first image now has the same display address as that of the pixel in the first image. The repositioned second digital image is then stored to memory.

20
25 In other embodiments the image area that is to be searched within the second image may be selected by a user of the method. In another embodiment an alignment point can be selected for aligning the first digital image and the repositioned second digital image such that the pixel of the first frame is aligned with the alignment point and the pixel in the second digital image is aligned with the alignment point when displayed on a display device.

30 Once the second digital image is saved the first digital image and the repositioned second digital image may be displayed sequentially on a display device

wherein the movement of the item will be minimized.

5 In a further embodiment, interpolation may be employed to determine color values for comparison in a sequence of video images. In the sequence, a beginning digital image and an ending digital image from the sequence creating a subset sequence are selected. A pixel is selected in the beginning digital image and a pixel is selected in the ending digital image. Linear interpolation is performed between the color of the pixel in the beginning digital image and the color of the pixel in the ending digital image such that an interpolated color is determined for each image between the beginning digital image and the ending digital image. For each interpolated color associated with an image, a pixel is searched for within that image within a predetermined range of colors. Once a pixel is found, each image is repositioned so that the address of the pixel having the interpolated color is positioned to the address of the pixel selected in the beginning image.

15 In certain embodiments, the color match range may be equal to zero, such that an exact match needs to be found. The greater the number of representative colors, such as true color or 16.7 million colors, the greater the likelihood that a match of one pixel in a first frame represents the same physical location as that of a pixel in a second frame having the same or a very similar color. If a color is not found in an image that is within a color match range, the method can default to a standardized formula for repositioning the image based upon historic information.

20 The method may also be implemented in a computer system and embodied on a computer program product for use with the computer system in which the method is embodied in computer readable code. Further, the method may be embodied in a system which includes a user interface, a processor, and associated memory. The processor includes a user interface module for receiving signals from the user interface. In such a configuration, a user may send signals to the processor through the user interface indicating the video sequence to be processed, indicating a color match range, selecting pixel values, selecting alignment positions, for example. The processor also includes a location module for locating a pixel within a second frame having an associated color that is within a color range of a color associated with the pixel selection from the first video frame. The processor further includes a repositioning

module for readdressing the second video frame so that the located pixel of the second frame has an address that is the same address as the pixel selection from the first video frame and also includes a storage module for storing the readdressed second video frame.

5

Brief Description of the Drawings

10 The features of the invention will be more readily understood by reference to the following detailed description, taken with reference to the accompanying drawings, in which:

Fig. 1 is an image having a resolution of 800x600 pixels;

15 Fig. 2 is a flow chart showing one method for image stabilization through color matching performed in conjunction with a computer system;

Figs 3A-C show pixels being repositioned on a screen;

Fig. 3A shows a first frame with a selected pixel;

Fig. 3B shows a second frame with a pixel marked having a color within a color tolerance of the pixel selected in Fig. 3A;

20 Fig. 3C shows the second frame of Fig. 3B superimposed and realigned on the first frame of Fig. 3A;

Fig. 4 is a flow chart showing another embodiment for image stabilization;

Fig. 5 is a flow chart showing another embodiment of image stabilization using color matching and interpolation;

25 Fig. 6 shows pixel selection in a beginning and ending frame of a video sequence;

Fig. 7A shows a video sequence prior to image stabilization in which all of the images are aligned;

Fig 7B shows the video sequence of Fig. 7A after realignment; and

30 Fig. 8 shows one embodiment of the invention as a system.

Detailed Description of Specific Embodiments

As used in the following text, the term "video sequence" implies a digital representation of a sequence of images which may be temporally displayed on a display device. Typically, a video sequence contains a plurality of frames wherein each frame represents a separate image. The frames may be further subdivided such that the frames are made up of a series of pixels. As used in the following description the term "pixel" shall mean a single point of an image. The greater the number of pixels that are contained in an image, the greater the resolution of the video. Resolution of an image/frame is conventionally referenced by length and width measurements of the number of pixels, for example, in a resolution of 800x600, there are 800 pixels along the length of an image by 600 pixels along the width of the image. Associated with each pixel is an address. For example as shown in Fig. 1. an image having a resolution of 800x600 is presented and pixel at address (200,300) is marked with an X wherein the origin is located at the upper left hand corner of the screen. As used herein a frame may be either a complete image or a portion of an image, such as a field which includes only one half of a complete image.

In an embodiment of the invention, a method is presented for stabilizing video images of a digital video sequence wherein the digital video sequence includes a plurality of digital images/frames. Stabilization is performed through color matching of a selected pixel color in a first frame by searching for a pixel in a second frame that has a color which falls within a color tolerance. The second frame is then aligned with the first frame such that the pixel in the second frame with the matching color has the same pixel address as that of the selected pixel from the first frame. The second frame is then stored to memory. This methodology is especially useful as the resolution and number of possible colors which can be represented increases. As video moves to HDTV formats which have 16.7 million and greater possible colors for a pixel color value, pixel color values become more unique and thus color tracking is more effective means for matching images and providing stabilization of images and accounting for motion within images.

The method for image stabilization through color matching is performed in conjunction with a computer system wherein the computer system includes at least a display, an input device, associated memory and a processor. The method is explained

in more formal detail with reference to Fig. 2. First a digital sequence is retrieved by the processor of the computer system from a memory location **200**. A color tolerance is either predetermined and accessed by the processor or the user is queried for entry of a color tolerance **210**. The color tolerance is the range within which a color match will be deemed to occur. A first frame of the video sequence is then displayed to a user. The user then selects a pixel within the first frame **220**. Preferably the pixel is of a distinct color or the pixel is part of an object which moves within the video sequence and appears jittery when the video sequence is displayed. After the pixel is selected, the processor identifies the color associated with the pixel **230**. The processor then retrieves the second frame and performs a search of the second frame to identify a pixel that has the same color as that of the selected pixel from the first frame within the color tolerance **240**. Color matching may be performed by taking the overall color for a pixel including all color components or weighting the individual color components. For example, if the video sequence is represented in an R,G,B, color space, an identical color match (i.e. one having a color tolerance of 0%) might require that each of the R,G, and B values are identical. The color tolerance may be the overall difference in color as a percentage. For example, a color tolerance may be .005% wherein the color space includes 16.7million colors or a color tolerance may require that each color component must be within a certain tolerance of say 10% of each R,G, and B value of the selected pixel. Once a pixel within the second frame is identified, the pixel address from the second frame is then aligned with the pixel address of the selected pixel of the first frame **250**. Figs. 3A, B and C show the pixels being repositioned. Fig. 3A shows a first frame with a selected pixel marked with an X as would be displayed in the display device. Fig. 3B is a second frame wherein a pixel that matches the selected pixel from the first frame is identified with an O. Fig. 3C shows that the displayed pixels of the second frame is repositioned, and thus readdressed. After the second frame is repositioned and thus the addresses of all of the pixels of the frame are changed, the realigned second frame is stored to memory by the processor **260**. This process can be continued for each frame of a video sequence. In such an embodiment, the color of the selected pixel from the first frame would be searched for in each subsequent frame. It should be noted that some pixel values of the realigned second frame will not be shown

on the display device because the new addresses of these pixels falls outside of the display addresses. Further, some display addresses for the second realigned frame will not have color values for the pixels. In one embodiment, these pixel addresses are made black. In other embodiments, default color values can be assigned such as the color values from adjacent pixel addresses.

The methodology may be further defined as shown in Fig. 4. As expressed above, a video sequence is received by a processor from a memory location. A first frame is presented to a user. The user defines a search area within a video frame step 401. For example, the user may use an input device such as a mouse or a light pen to graphically indicate the search area. In one embodiment, the entire video frame is the default search area. Defining a search area that is less than the entire image may be beneficial in images in which a user is interested in a particular element of an image. For example, if the video sequence is of a car that is driving away and the user is interested in the license plate of the car, the user may select a sub-section of the video frame. In the example, the user would indicate that the license plate or a portion of the car was the selected sub-set of the image. The image displayed after re-alignment of the frames could be the entire video frame or just the selected sub-set of the frame.

In a video sequence in which the moving object that is desired to be made still moves across the entire frame, selection of a sub-section of the image, such as that of a license plate, should be done over a short sequence of frames. For example, if the entire sequence is of 120 fields of video (approx. 2sec.), the sequence may be subdivided such that only a few frames (for example 10 fields) are used. The car and the license plate can then be designated as the sub-section of the image for comparison purposes since the license plate of the car will not move many pixel locations over 10 fields. By selecting a sub-section, this improves the odds that the color matched pixels actually represent the same physical point.

The user is also prompted by the computer system for a color tolerance which as explained above is the acceptable margin of error for matching a color 402. The user can also indicate an alignment point which is the point on the output screen of the display device at which to align the pixel whose color best matches the tracking color 403. In one embodiment, the default is the center of the screen of the display device

although other positions on the display may also be selected. The user also provides the color to be tracked **404**. This is done by observing the first video frame and selecting a pixel using the user input device.

5 For each video frame **420**, the color values for each pixel of the search area are compared **430** with the color values of the selected tracking color from the first video frame. If the color values for the color space (e.g. R,G,B) of a pixel are within the entered tolerance **440**, the pixel will be considered a match. The pixel in which the sum of the absolute values of the differences between the red, green and blue color values to those of the tracking color is smallest is identified as the color match point. If
10 the color match point is within the entered tolerance for each of the color space values associated with the selected pixel, the frame is shifted **450** so that the pixel identified as the best color match is placed at the alignment point on the screen and the shifted frame is then written out to memory. If no color match is found, the image is shifted **460** according to a predetermined formula. For instance, the shifts may be determined by
15 analyzing past shifts and averaging the shifts to determine how to align the frame of video data. For example, the image may be shifted in the x direction by a weighted average of previously shifted distances in the x-direction. Similarly, the shift in the y direction may be a weighted average of previously shifted distances in the y-direction.

20 In other embodiments, the image may be shifted according to the following formulae:

$x_{diff} = ((sx1 - sx2) + (sx2 - sx3) + (sx3 - sx4)) / 3 * (sd);$ and

$y_{diff} = ((sy1 - sy2) + (sy2 - sy3) + (sy3 - sy4)) / 3 * (sd);$ where

$sx1$ = previous horizontal shift; $sy1$ = previous vertical shift;

$sx2$ = horizontal shift before $sx1$; $sy2$ = vertical shift before $sy1$;

25 $sx3$ = horizontal shift before $sx2$; $sy3$ = vertical shift before $sy2$;

$sx4$ = horizontal shift before $sx3$; $sy4$ = vertical shift before $sy3$; and

where sd is a slowdown factor.

The frame is then shifted **460** by an amount given by the following equations:

horizontal shift = $sx1 + x_{diff}$; and

30 vertical shift = $sy1 + y_{diff}$.

The shifted frame is then written out **465** to the video image store.

If the input stream contains **470** more video frames, the next frame is processed
420. Once all of the frames in the input stream are processed, the color tracking
operation is complete.

Once all of the frames of video are repositioned using the color tracking, the
sequence of frames may be temporally displayed on a display device and thus
movement of the camera or object will be reduced. Other than display of the video
frames in a temporal fashion, the video frames may be linearly averaged in order to
provide a higher definition image or portion of an image. For example, a license plate
on a vehicle that is moving and which is unreadable due to jitter may be readable once
the image is stabilized. Even if it is not, all of the frames may be linearly averaged
together to produce a higher resolution image that may provide an enhanced image
wherein the license plate is legible. By precisely aligning the images to a sub-pixel
resolution, and averaging the images together, a higher resolution image can be
generated. This is due to the fact that as several images of an object are captured, each
image presents a slightly different part of the object in each frame. In general a
minimum of three to five frames are needed to improve the image.

In another embodiment of the invention, interpolation is employed in
conjunction with color matching as is explained in Fig. 5. In such an embodiment, a
video sequence is provided into a computer system **500**. A user of the system is allowed
to view the video sequence on a display device and select a beginning frame and an
ending frame **501**. A pixel is selected in both the beginning frame and the ending frame
502. Preferably the pixel in the beginning frame and the ending frame is a pixel is part
of an object for which motion stabilization is desired. For instance, a pixel may be
selected from a video image of a car that is moving. A pixel in the ending frame should
be selected that is part of the car as shown in Fig. 6. The system then interpolates
between the color value for the selected pixel in the beginning frame and the color
value of the pixel in the ending frame **503**. The interpolation method may be any
interpolation method known by one of ordinary skill in the art, such as linear, bi-linear,
and bi-cubic interpolation. The interpolation method provides interpolation values for
each frame between the beginning frame and the ending frame. For example, if there
are four frames, an interpolated color value would be determined for the second and

third frames. Once a color value is determined for each frame between the beginning and ending frames, the processor begins to search for that color value in each respective frame **504**. The processor may use a color match range/color tolerance to locate a pixel that matches the color. In one embodiment the processor searches for the first pixel that has the exact color and does not use a color tolerance. In another embodiment, each pixel that is within the color tolerance is saved to memory and a difference measurement as described above is determined until the pixel with the closest color match is determined **505**. If no pixels match, the processor employs a formula such as that provided above, or another formula that is based upon trend analysis or another property of the video sequence **506**. When a pixel has been selected for each frame, the frames after the beginning frame including the ending frame are re-aligned **507**. Realignment requires that the associated addresses of each pixel are re-determined, such that when the video sequence is displayed on a display device the pixel location of the pixel selected in the first frame corresponds with the pixels in the subsequent frames that match the color value. For example, as shown in Fig. 7A a pixel is selected in the beginning frame wherein the remaining frames are aligned as they would normally be displayed. Fig 7B shows the video frames realigned such that the color matched pixels from all frames subsequent to the beginning frame are positioned at the same point on a display screen as the selected pixel from the beginning frame. This process can be repeated if less than the entire video sequence is included between the beginning frame and the ending frame.

The method for image stabilization may be embodied as a system as shown in Fig. 8. The system **800** includes an input device **801** such as a mouse, trackball, keyboard, etc. and video display **802** and a processor **803** having associated memory **804**. The processor includes various modules. The first module is a user interface module **803A** that operates to receive signals from the user input device. The user can input a request to perform image stabilization on a selected video image sequence. The user can view the first frame of the video sequence and select a pixel within the sequence wherein the user interface module **803A** receive this information and sends the information to a location module **803B** that identifies the color of the selected pixel and accesses a second frame of the video data. The location module **803B** searches the

data values of the second frame in order to identify the pixel that is the closest color match to the pixel selected in the first frame. If no pixel is found that is within the color range, then the location module **803B** defaults to a predetermined formula such as that which was described above. When a pixel has been identified, the location of the pixel is presented to a repositioning module **803C** that re-addresses all of the pixels within the second frame such that the located pixel in the second frame has a new address that is the same display address as the pixel selected in the first frame. The address information along with the corresponding pixel color values are stored to memory by a storage module **803D**.

It should be noted that the flow diagrams are used herein to demonstrate various aspects of the invention, and should not be construed to limit the present invention to any particular logic flow or logic implementation. The described logic may be partitioned into different logic blocks (e.g., programs, modules, functions, or subroutines) without changing the overall results or otherwise departing from the true scope of the invention. Oftentimes, logic elements may be added, modified, omitted, performed in a different order, or implemented using different logic constructs (e.g., logic gates, looping primitives, conditional logic, and other logic constructs) without changing the overall results or otherwise departing from the true scope of the invention.

The present invention may be embodied in many different forms, including, but in no way limited to, computer program logic for use with a processor (e.g., a microprocessor, microcontroller, digital signal processor, or general purpose computer), programmable logic for use with a programmable logic device (e.g., a Field Programmable Gate Array (FPGA) or other PLD), discrete components, integrated circuitry (e.g., an Application Specific Integrated Circuit (ASIC)), or any other means including any combination thereof.

Computer program logic implementing all or part of the functionality previously described herein may be embodied in various forms, including, but in no way limited to, a source code form, a computer executable form, and various intermediate forms (e.g., forms generated by an assembler, compiler, linker, or locator.) Source code may include a series of computer program instructions implemented in any of various programming languages (e.g., an object code, an assembly language, or a high-level

language such as Fortran, C, C++, JAVA, or HTML) for use with various operating systems or operating environments. The source code may define and use various data structures and communication messages. The source code may be in a computer executable form (*e.g.*, via an interpreter), or the source code may be converted (*e.g.*, via a translator, assembler, or compiler) into a computer executable form.

The computer program may be fixed in any form (*e.g.*, source code form, computer executable form, or an intermediate form) either permanently or transitorily in a tangible storage medium, such as a semiconductor memory device (*e.g.*, a RAM, ROM, PROM, EEPROM, or Flash-Programmable RAM), a magnetic memory device (*e.g.*, a diskette or fixed disk), an optical memory device (*e.g.*, a CD-ROM), a PC card (*e.g.*, PCMCIA card), or other memory device. The computer program may be fixed in any form in a signal that is transmittable to a computer using any of various communication technologies, including, but in no way limited to, analog technologies, digital technologies, optical technologies, wireless technologies, networking technologies, and internetworking technologies. The computer program may be distributed in any form as a removable storage medium with accompanying printed or electronic documentation (*e.g.*, shrink wrapped software or a magnetic tape), preloaded with a computer system (*e.g.*, on system ROM or fixed disk), or distributed from a server or electronic bulletin board over the communication system (*e.g.*, the Internet or World Wide Web.)

Hardware logic (including programmable logic for use with a programmable logic device) implementing all or part of the functionality previously described herein may be designed using traditional manual methods, or may be designed, captured, simulated, or documented electronically using various tools, such as Computer Aided Design (CAD), a hardware description language (*e.g.*, VHDL or AHDL), or a PLD programming language (*e.g.*, PALASM, ABEL, or CUPL.)

The present invention may be embodied in other specific forms without departing from the true scope of the invention. The described embodiments are to be considered in all respects only as illustrative and not restrictive.

01247/00A65 265929.1